深圳开源通信有限公司

OpenVox A400P/A400E 用户手册

深圳开源通信有限公司

OpenVox-Best Cost Effective Asterisk Cards

广东省深圳市福田区滨河路上沙创新科技园 14 栋 2 楼
电话：0755－82535461
0755－83545095
0755－82535362
传真：0755－82535174
业务邮箱：sales@openvox.com.cn
技术支持邮箱：support@openvox.com.cn

上班时间是周一至周五早上 9：00－下午 6：00，节假日除外。（GMT＋8 北京时间）

Thank You for Choosing OpenVox Products!

目录

第一章 概述 4

第二章 硬件介绍 5

第三章 安装及配置 8

第四章 测试系统 20

第五章 参考 23

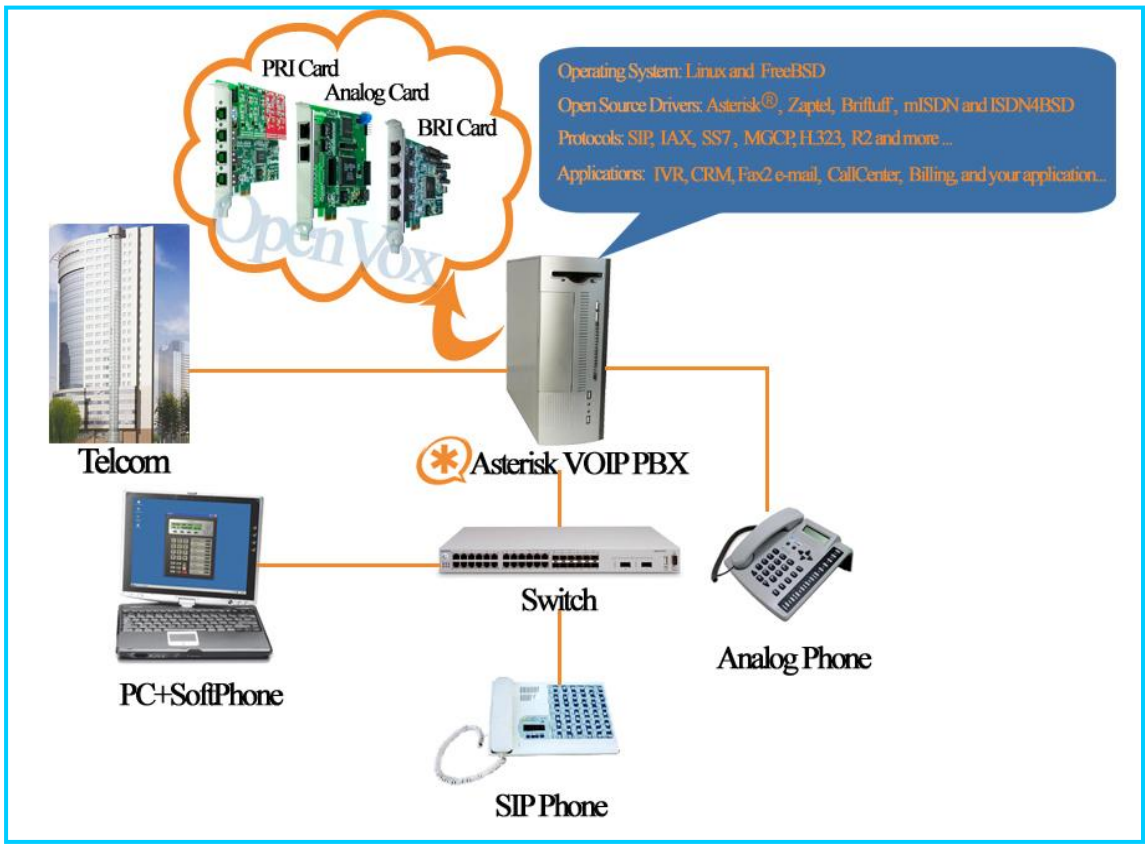
第一章 概述

1. A400P/A400E 是什么

A400P/A400E 是一个模块化的模拟电话接口产品，它是专为硬件兼容 Digium 公司的 TDM400p 而设计的。A400P/A400E 须使用 FXO-100 或 FXS-100 来组成一个可行的系统。FXO-100 和 FXS-100 模块同样兼容 X100M 和 S100M。

2. 什么是 Asterisk

Asterisk 是一个开放源代码的软件 VoIP PBX 系统，它是一个运行在 Linux 环境下的纯软件实施方案。Asterisk 是一种功能非常齐全的应用程序，提供了许多电信功能，能够把你的 x86 机器变成你自己的交换机，还能够当作一台企业级的商用交换机。Asterisk 让人激动的事情是它在小企业预算可承受的范围内提供了商业交换机的功能和可伸缩性。



Asterisk 的通道维护着 VoIP 和 PSTN 上的各种形式的链接，包含的 VoIP 协议有 SIP,IAX,MGCP 和 H.323。

第二章　硬件介绍

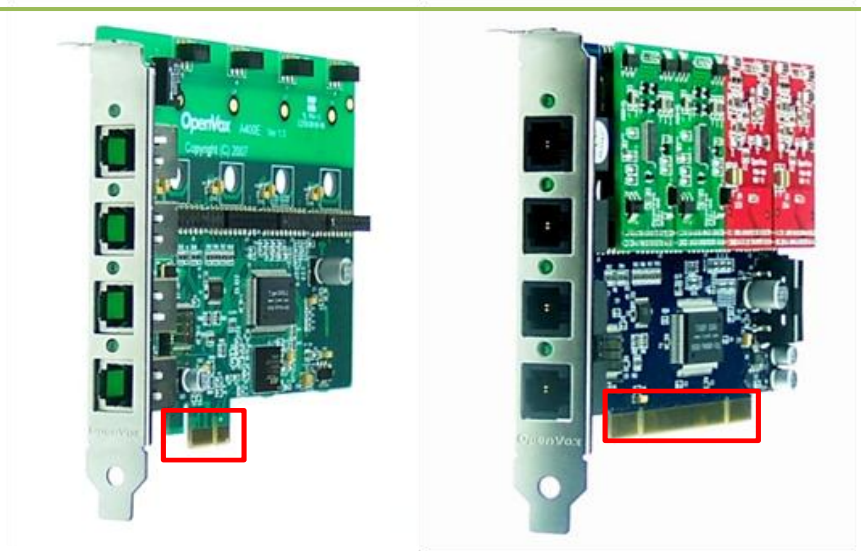
操作环境：

CentOS 5.0

zaptel-1.4.12.1

asterisk-1.4.21.1

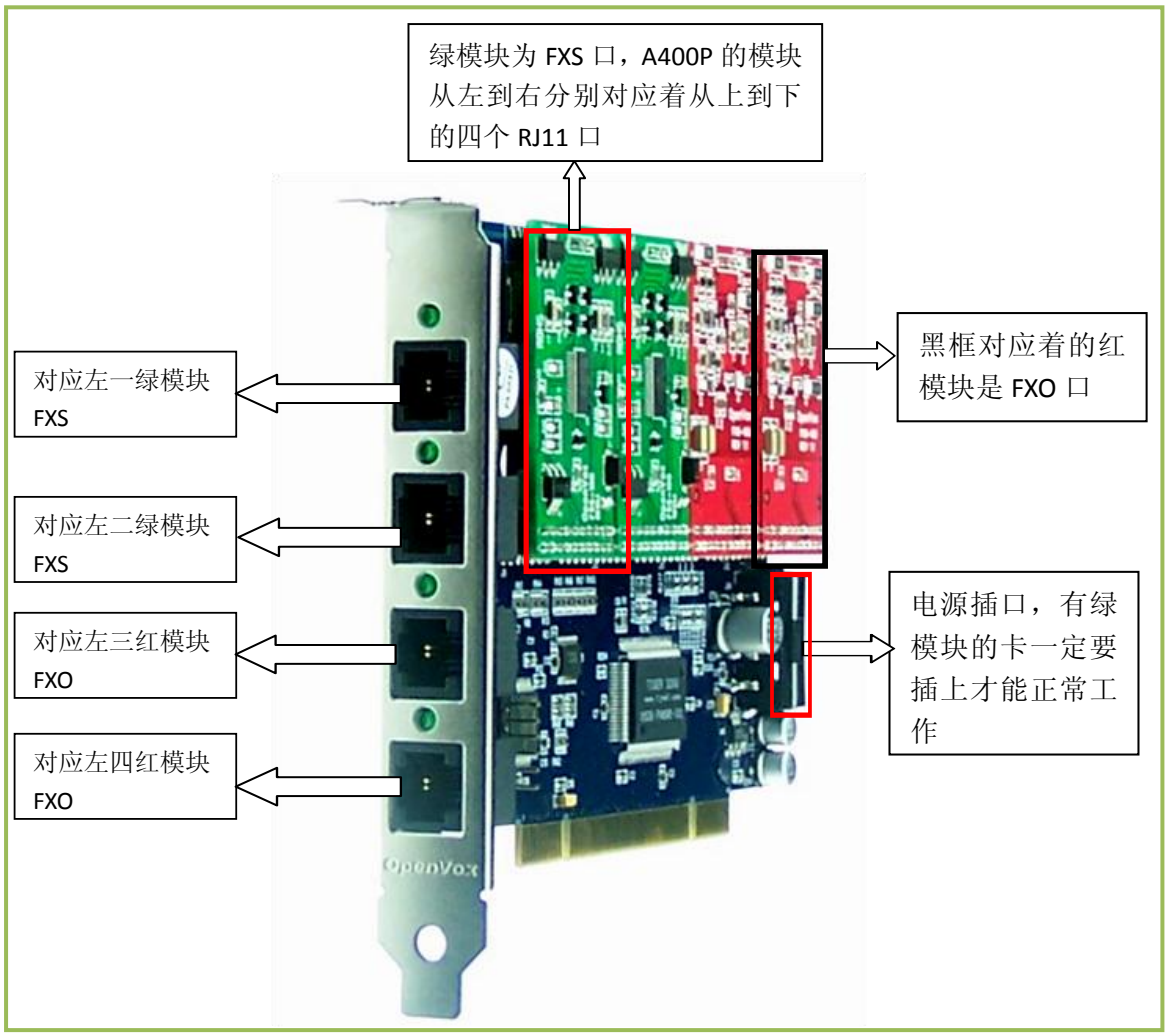
openvox A400P 2FXS + 2FXO

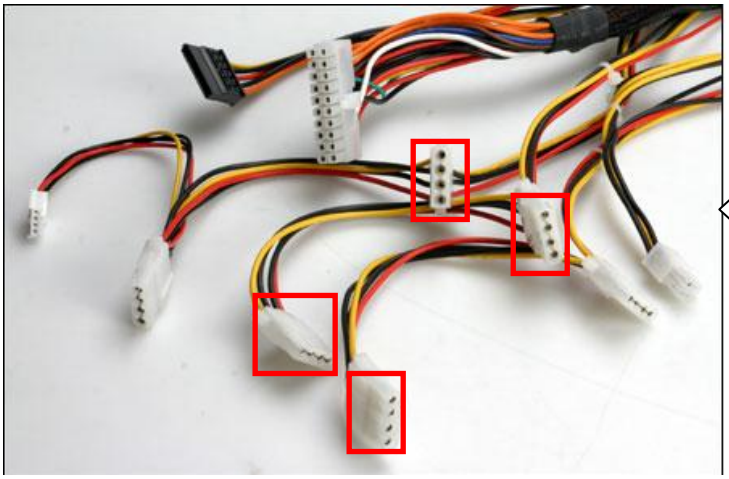


A400E　　A400P

A400P和A400E的唯一区别就是它们使用的插槽类型不一样，A400P对应的是PCI插槽，而A400E对应的是PCIE插槽，如上图红框所标注的

A400P 产品样图：

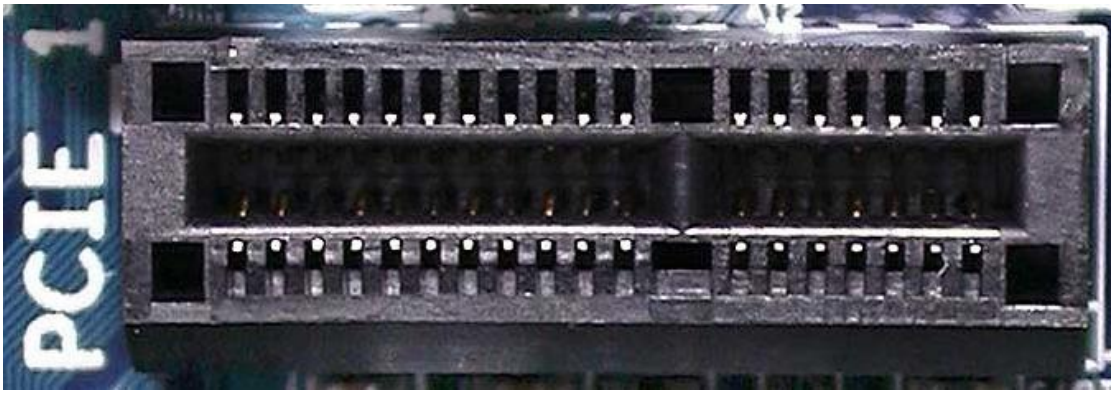

绿模块为 FXS 口，A400P 的模块从左到右分别对应着从上到下的四个 RJ11 口
对应左一绿模块 FXS
对应左二绿模块 FXS
对应左三红模块 FXO
对应左四红模块 FXO
黑框对应着的红模块是 FXO 口
电源插口，有绿模块的卡一定要插上才能正常工作



用红框标出的这几个 PC 电源插口就是 A400P /A400E 的电源插口，对应的上图所标出的电源插口

一块主板上的32位 PCI 扩充插槽

这几个白色的插槽就是主板上的 PCI 插槽，A400P 就是插入这种插槽



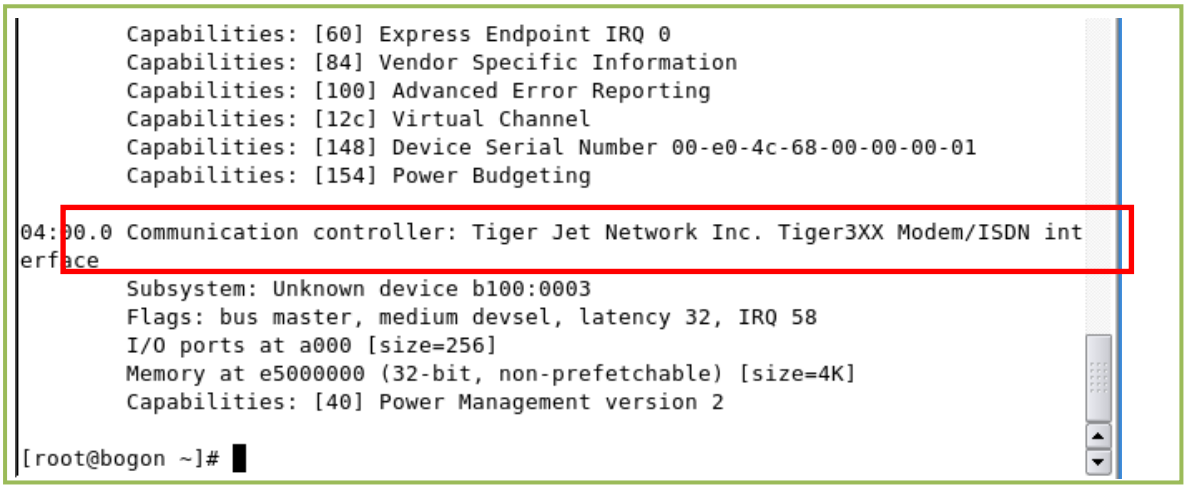
上图这个黑色的插槽就是主板上的 PCI-E 插槽，A400E 就是插入这种插槽

第三章 安装及配置

一．把 A400P 的卡插入 PCI 插槽，插好电源（如果你有绿模块的话）

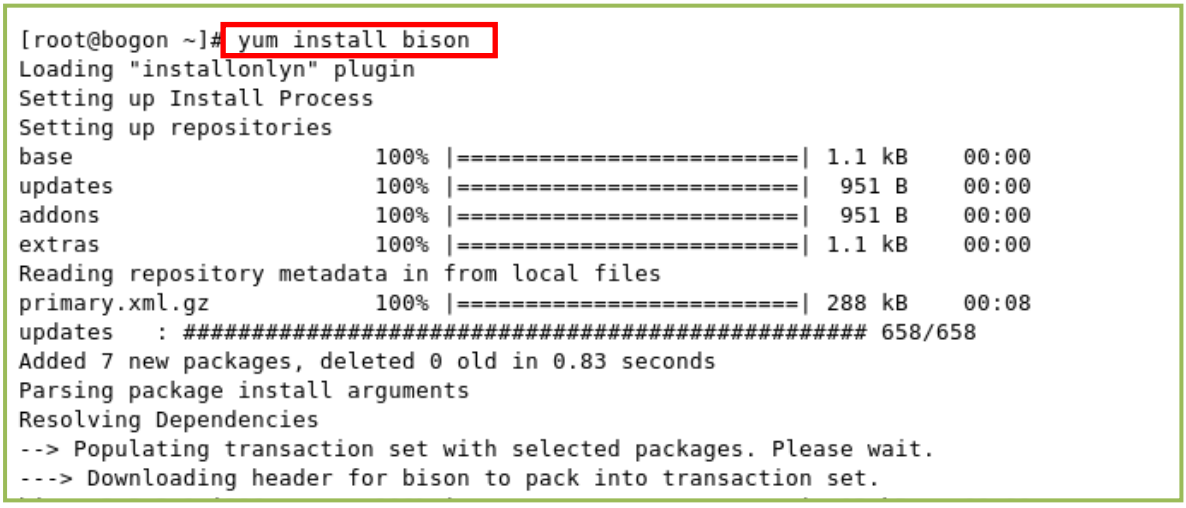
二．启动 CentOS 5.0，进入系统，在命令行下输入 lspci –v ，如图：

检查输出结果，查看系统是否识别 A400P，如下图红框标出表示识别：



三．检查安装 asterisk 所必备的软件包，如图：

以上输出表示系统没有安装这个软件包，用 *yum install xxxx* 命令安装，如图：



安装 asterisk 所必需的软件包总共有十一个，如下分别用 *rpm –q xxxx* 命令检查：

```
rpm   -q   bison
rpm   -q   bison-devel
rpm   -q   ncurses
rpm   -q   ncurses-devel
rpm   -q   zlib
rpm   -q   zlib-devel
rpm   -q   openssl
rpm   -q   openssl-devel
rpm   -q   gnutls-devel
rpm   -q   gcc
rpm   -q   gcc-c++
```

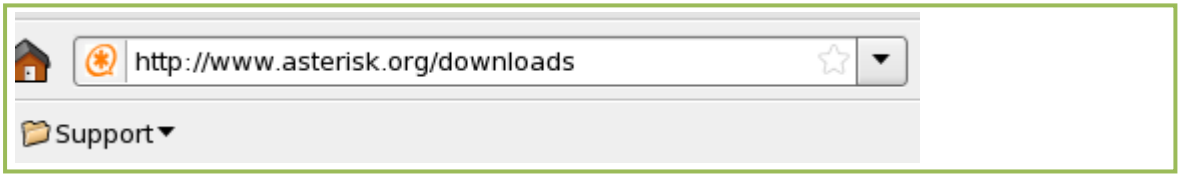
如果遇到没有安装便如上图所示使用 *yum install xxxx* 命令安装即可，确保以上所列出的安装包都要装上。//xxxx 代表对应包

四．接下来就是要下载 zaptel 和 asterisk

用 firefox 打开 www.asterisk.org，下载相同版本的 zaptel 和 asterisk，建议是 1.4 的版本。

下载方法一：

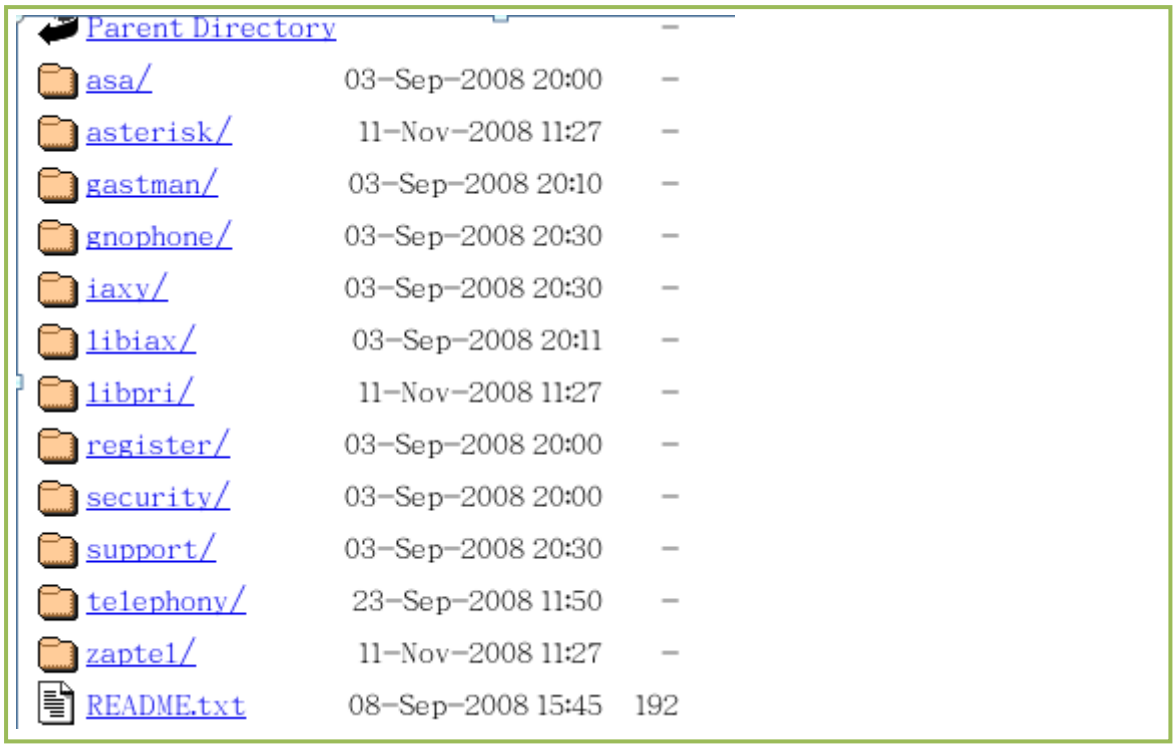
1) 进入网页，如图：



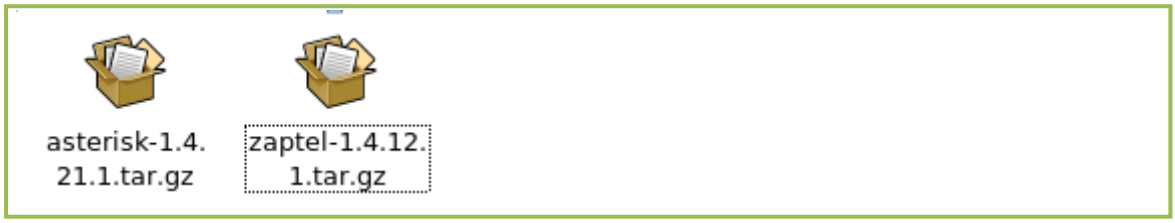
2) 找到网页上的[http]这个下载标志，图中用红框标出：

Bandwidth for Asterisk.org and AsteriskNOW.org sponsored by API Digital and Bandwidth.com.
Complete directory
downloads.digium.com - [http]
AsteriskNOW project
Asterisk made easy
Not familiar enough with Linux to install Asterisk via a command line interface? Want Asterisk made easy?
Asterisk can now be easily configured with a graphical interface. AsteriskNOW™ is an open source
and Digital interface cards. Digium hardware requires Zaptel drivers.
Other projects
Libiax
Library for the Inter-Asterisk eXchange (IAX) protocol. Libiax provides a foundation for the development of IAX clients.

3) 点击进入，找到 zaptel 和 asterisk 文件夹，分别点击进去下载所对应版本的包到 /usr/src/目录下，如图：



4) 下载后的包， 如图：



下载方法二：

在命令行中输入 *cd /usr/src* 进入 src 目录下，在输入 wget + 文件地址即可下载

五．接下来就是编译 zaptel 和 asterisk ，具体操作如下：

1) 输入

cd /usr/src

tar -xvzf zaptel-xxxx //解压包 xxxx 代表对应的版本号

tar -xvzf asterisk-xxxx //解压包 xxxx 代表对应的版本号

如图：

```
[root@bogon src]# tar -xvzf zaptel-1.4.12.1.tar.gz

[root@bogon src]# tar -xvzf asterisk-1.4.21.1.tar.gz
```

输入

cd zaptel-xxxx

./configure

如图：

```
[root@bogon src]# cd zaptel-1.4.12.1
[root@bogon zaptel-1.4.12.1]# ./configure
```

下图是*./configure* 运行正确完成的输出：

```
checking for initscr in -lncurses... yes
checking for curses.h... (cached) yes
checking for newtBell in -lnewt... yes
checking newt.h usability... yes
checking newt.h presence... yes
checking for newt.h... yes
checking for usb_init in -lusb... no
configure: creating ./config.status
config.status: creating build_tools/menuselect-deps
config.status: creating makeopts
config.status: creating build_tools/make_firmware_object
configure: *** Zaptel build successfully configured ***
[root@bogon zaptel-1.4.12.1]#
```

输入 *make*

```
root@bogon zaptel-1.4.12.1]#
root@bogon zaptel-1.4.12.1]# make
```

下图是 *make* 命令正确完成的输出：

```
pod2man --section 8 zt_registration > zt_registration.8 || rm -f zt_registratior
.8
pod2man --section 8 xpp_sync > xpp_sync.8 || rm -f xpp_sync.8
pod2man --section 8 lszaptel > lszaptel.8 || rm -f lszaptel.8
pod2man --section 8 xpp_blink > xpp_blink.8 || rm -f xpp_blink.8
pod2man --section 8 zapconf > zapconf.8 || rm -f zapconf.8
pod2man --section 8 zaptel_hardware > zaptel_hardware.8 || rm -f zaptel_hardware
.8
make[2]: Leaving directory `/usr/src/zaptel-1.4.12.1/kernel/xpp/utils'
make[1]: Leaving directory `/usr/src/zaptel-1.4.12.1'
[root@bogon zaptel-1.4.12.1]#
```

输入 *make install*

```
[root@bogon zaptel-1.4.12.1]#
[root@bogon zaptel-1.4.12.1]#
[root@bogon zaptel-1.4.12.1]# make install
```

下图是 *make install* 命令正确完成的输出：

```
Firmware zaptel-fw-oct6114-128.bin is already installed with required version 1.
05.01
Firmware zaptel-fw-tc400m.bin is already installed with required version MR6.12
Firmware zaptel-fw-vpmadt032.bin is already installed with required version 1.07
make[1]: Leaving directory `/usr/src/zaptel-1.4.12.1/firmware'
################################################
###
### Zaptel installed successfully.
### If you have not done so before, install init scripts with:
###
###   make config
###
################################################
[root@bogon zaptel-1.4.12.1]#
```

输入 *make config*

```
[root@bogon zaptel-1.4.12.1]#
[root@bogon zaptel-1.4.12.1]#
[root@bogon zaptel-1.4.12.1]# make config
```

下图是 *make config* 命令正确完成的输出：

```
If you have any zaptel hardware it is now recommended to
edit /etc/default/zaptel or /etc/sysconfig/zaptel and set there an
optimal value for the variable MODULES .

I think that the zaptel hardware you have on your system is:
pci:0000:04:00.0     wctdm+     e159:0001 Wildcard TDM400P REV E/F
```

现在已经编译完 zaptel ， 接下来开始编译 asterisk，命令和编译 zaptel 的基本一致：

输入 *cd /usr/src/asterisk-xxxx //xxxx 代表对应的版本号*

./configure

```
[root@bogon zaptel-1.4.12.1]#
[root@bogon zaptel-1.4.12.1]# cd ..
[root@bogon src]# cd asterisk-1.4.21.1
[root@bogon asterisk-1.4.21.1]# ./configure
```

下图是 *./ configure* 命令正确完成的输出：

```
.$$7          .7$$$$$$$7:          ?$$$.
$$$          ?7$$$$$$$$$$I        .$$$7
$$$       .7$$$$$$$$$$$$$$$      :$$$.
$$$       $$$$$$7$$$$$$$$$$$    .$$$.
$$$        $$$   7$$$7  .$$$    .$$$.
$$$$             $$$$7         .$$$.
7$$$7            7$$$$        7$$$
 $$$$$                        $$$
  $$$$7.                       $$  (TM)
   $$$$$$$.           .7$$$$$$  $$
     $$$$$$$$$$$$$7$$$$$$$$.$$$$$$
       $$$$$$$$$$$$$$$$.

configure: Package configured for:
configure: OS type  : linux-gnu
configure: Host CPU : i686
[root@bogon asterisk-1.4.21.1]#
```

输入 *make*

```
root@bogon asterisk-1.4.21.1]#
root@bogon asterisk-1.4.21.1]#
root@bogon asterisk-1.4.21.1]# make
```

下图是 *make* 命令正确完成的输出：

```
cno/rec_utils.o -> libdb1.a
   [LD] abstract_jb.o acl.o aescrypt.o aeskey.o aestab.o alaw.o app.o ast_expr2.
o ast_expr2f.o asterisk.o astmm.o astobj2.o audiohook.o autoservice.o callerid.o
 cdr.o channel.o chanvars.o cli.o config.o cryptostub.o db.o devicestate.o dial.
o dns.o dnsmgr.o dsp.o enum.o file.o fixedjitterbuf.o frame.o fskmodem.o global_
datastores.o http.o image.o indications.o io.o jitterbuf.o loader.o logger.o man
ager.o md5.o netsock.o pbx.o plc.o privacy.o rtp.o say.o sched.o sha1.o slinfact
ory.o srv.o stdtime/localtime.o strcompat.o tdd.o term.o threadstorage.o transla
te.o udptl.o ulaw.o utils.o editline/libedit.a db1-ast/libdb1.a -> asterisk
 +--------- Asterisk Build Complete ---------+
 + Asterisk has successfully been built, and +
 + can be installed by running:              +
 +                                           +
 +               make install                +
 +-------------------------------------------+
[root@bogon asterisk-1.4.21.1]#
```

输入 *make install*
运行后得到的正确结果如图：

```
+---- Asterisk Installation Complete -------+
+                                            +
+    YOU MUST READ THE SECURITY DOCUMENT     +
+                                            +
+ Asterisk has successfully been installed.  +
+ If you would like to install the sample    +
+ configuration files (overwriting any       +
+ existing config files), run:               +
+                                            +
+                make samples                +
+                                            +
+-----------------  or ----------------------+
+                                            +
+ You can go ahead and install the asterisk  +
+ program documentation now or later run:    +
+                                            +
+               make progdocs                +
+                                            +
+ **Note** This requires that you have       +
+ doxygen installed on your local system     +
+--------------------------------------------+
root@bogon asterisk-1.4.21.1]#
```

输入 *make samples*

```
[root@bogon asterisk-1.4.21.1]#
[root@bogon asterisk-1.4.21.1]#
[root@bogon asterisk-1.4.21.1]# make samples
```

下图是 *make samples* 命令正确完成的输出：

```
 channel is being recorded or DTMF is being generated" ; \
                echo ";transcode_via_sln = yes ; Build transcode paths via SL
AR, instead of directly" ; \
                echo ";runuser = asterisk ; The user to run as" ; \
                echo ";rungroup = asterisk ; The group to run as" ; \
                echo "" ; \
                echo "; Changing the following lines may compromise your secu
y." ; \
                echo ";[files]" ; \
                echo ";astctlpermissions = 0660" ; \
                echo ";astctlowner = root" ; \
                echo ";astctlgroup = apache" ; \
                echo ";astctl = asterisk.ctl" ; \
                ) > /etc/asterisk/asterisk.conf ; \
        else \
                echo "Skipping asterisk.conf creation"; \
        fi
mkdir -p /var/spool/asterisk/voicemail/default/1234/INBOX
build_tools/make_sample_voicemail //var/lib/asterisk //var/spool/asterisk
[root@bogon asterisk-1.4.21.1]#
```

六．编译完上述两个文件之后，再运行下面的命令：

cd /usr/src/zaptel-xxxx/kernel/xpps/utils //xxxx 代表对应的版本号

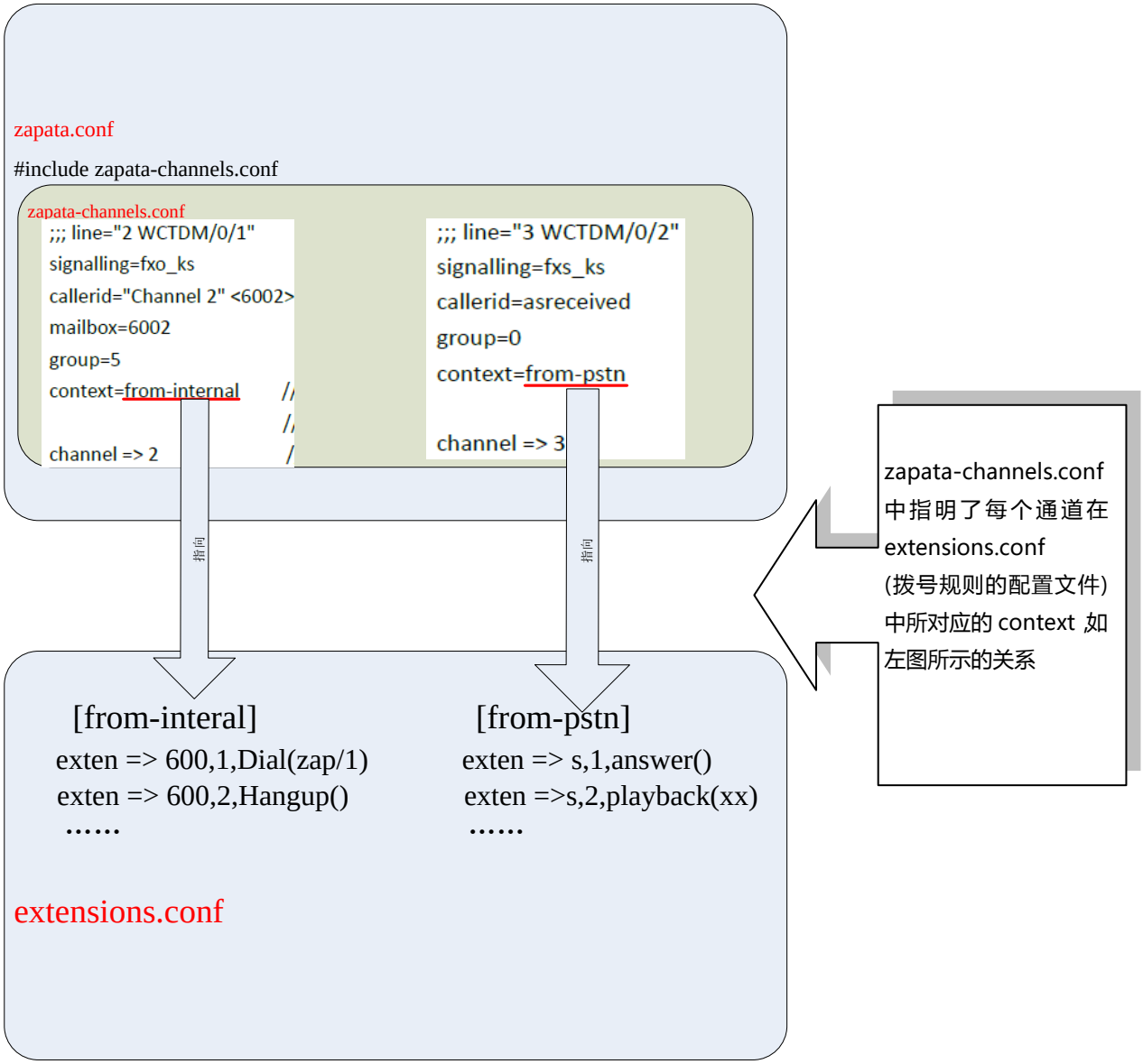
./genzaptelconf -sdvM

（运行这个命令后，如果上面有报 error，请认真查看报错的具体信息，系统会自动给出

解决问题的命令，用户可以根据该命令进行解决，然后再次运行./genzaptelconf -sdvM ,如无报错，则可进行下一步）

这个命令会自动生成/etc/zaptel.conf 和 /etc/asterisk/zapata-channels.conf 文件，以下是生成的两个文件：

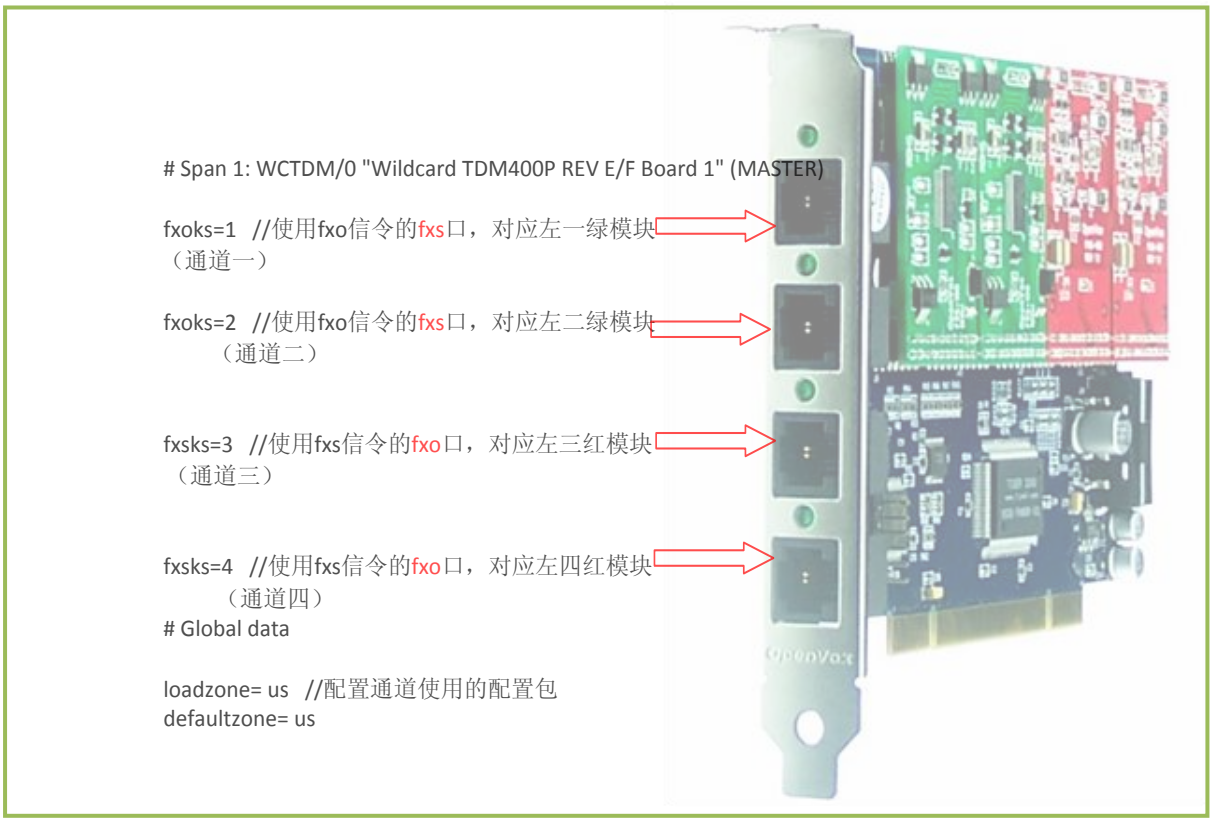
zapata.conf , zapata-channels.conf, extensions.conf 三者的关系图：



zaptel.conf 文件如下：

(zaptel 文件是 A400P 卡的基本通道配置文件)

```
# Autogenerated by ./genzaptelconf -- do not hand edit
# Zaptel Configuration File
# This file is parsed by the Zaptel Configurator, ztcfg
# It must be in the module loading order
```



/etc/asterisk/zapata-channels.conf 文件如下：（用蓝色框住的部分)

```
; Autogenerated by ./genzaptelconf -- do not hand edit
; Zaptel Channels Configurations (zapata.conf)
;
; This is not intended to be a complete zapata.conf. Rather, it is intended
; to be #include-d by /etc/zapata.conf that will include the global settings
;
; Span 1: WCTDM/0 "Wildcard TDM400P REV E/F Board 1" (MASTER)
;;; line="1 WCTDM/0/0"
signalling=fxo_ks       //定义信令方式为 FXO
callerid="Channel 1" <6001>     //定义主叫名字和号码
mailbox=6001        //定义语音信箱号码
group=5

context=from-internal    //定义拨号规则的 context ,必须
                         //  与/etc/asterisk/extensions.conf 的 context 相对应
channel => 1              //通道一
callerid=
mailbox=
group=
context=default
```

```
;;; line="2 WCTDM/0/1"
signalling=fxo_ks
callerid="Channel 2" <6002>
mailbox=6002
group=5
context=from-internal      //定义拨号规则的 context ,必需要
                           //  与/etc/asterisk/extensions.conf 的 context 相对应
channel => 2               //通道二
callerid=
mailbox=
group=
context=default

;;; line="3 WCTDM/0/2"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn          //定义拨号规则的 context ,必需要
                           //  与/etc/asterisk/extensions.conf 的 context 相对应
channel => 3               //通道三
context=default

;;; line="4 WCTDM/0/3"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn          //定义拨号规则的 context ,必需要
                           //  与/etc/asterisk/extensions.conf 的 context 相对应
channel => 4               //通道四
context=default
```

检查自动生成的文件配置是否和自己的 A400P 的卡一致，如果不一致的话要进行修改，注意 FXS 口用 FXO 信令，FXO 口用 FXS 信令

```
[root@bogon asterisk-1.4.21.1]#
[root@bogon asterisk-1.4.21.1]#
[root@bogon asterisk-1.4.21.1]# cd /usr/src/zaptel-1.4.12.1/kernel/xpp/utils/
```

```
[root@bogon utils]#
[root@bogon utils]#
[root@bogon utils]# ./genzaptelconf -sdvM
```

下图为运行上述命令的输出，从输出我们可以发现一个问题，（使用本文的所用的软件版本可能会出现的一个问题，其他的版本不一定会出现），/etc/asterisk/zapata-channels.conf 没有被包含进 /etc/asterisk/zapata.conf，解决办法是：

在 zapata.conf 文件的最后加一句： *#include zapatal-channels.conf* 即可：

```
Updating modules list in zaptel init config /etc/sysconfig/zaptel.
Generating '/etc/zaptel.conf and /etc/asterisk/zapata-channels.conf'
Note: generated /etc/asterisk/zapata-channels.conf not included in zapata.conf
To fix:  echo '#include zapata-channels.conf' >>/etc/asterisk/zapata.conf
Reconfiguring identified channels

Zaptel Version: 1.4.12.1
Echo Canceller: MG2
Configuration
=====================


Channel map:

Channel 01: FXO Kewlstart (Default) (Slaves: 01)
Channel 02: FXO Kewlstart (Default) (Slaves: 02)
Channel 03: FXS Kewlstart (Default) (Slaves: 03)
Channel 04: FXS Kewlstart (Default) (Slaves: 04)

4 channels to configure.
```

输入

cd /etc/asterisk/

vi zapata.conf

```
extconfig.conf        musiconhold.conf.old  vpb.conf
extensions.ael        muted.conf            zapata-channels.conf
extensions.conf       osp.conf              zapata-channels.conf.bak
extensions.conf.old   oss.conf              zapata.conf
features.conf         phone.conf            zapata.conf.old
festival.conf         privacy.conf
[root@bogon asterisk]# cd /etc/asterisk/


[root@bogon asterisk]#
[root@bogon asterisk]#
[root@bogon asterisk]# vi zapata.conf
```

在文件最后添加一句：

#include zapata-channels.conf

```
;context=default
;channel => 1
#include zapata-channels.conf
```

接下来输入一下命令：

modprobe zaptel //加载 zaptel 模块

modprobe wctdm //加载 wctdm 模块

ztcfg -vvvvv

```
[root@bogon utils]#
[root@bogon utils]# modprobe zaptel
[root@bogon utils]# modprobe wctdm
[root@bogon utils]#

[root@bogon utils]# ztcfg -vvvvv

Zaptel Version: 1.4.12.1
Echo Canceller: MG2
Configuration
=====================


Channel map:

Channel 01: FXO Kewlstart (Default) (Slaves: 01)
Channel 02: FXO Kewlstart (Default) (Slaves: 02)
Channel 03: FXS Kewlstart (Default) (Slaves: 03)
Channel 04: FXS Kewlstart (Default) (Slaves: 04)

4 channels to configure.
```

上面的输出表示 A400P 的通道都已识别，驱动安装成功（本文用的的 2 个 FXS + 2 个 FXO 口，上面的 channel 01 和 channel 02 是 FXS 口，后面的 FXO 表示它们使用的是 FXO 的信令，并不是表示它们是 FXO 口，channel 03 和 channel 04 同理这一点大家应该注意）

七．启动 asterisk，运行下面命令：

asterisk -vvvvvvvgc (如果 asterisk 已启动，可输入 asterisk -r)

进入 CLI 后输入 *zap show channels* ,如果能看到模块表示安装已基本完成：

```
*CLI> zap show channels
   Chan Extension  Context         Language   MOH Interpret
 pseudo            default                    default
      1            from-internal              default
      2            from-internal              default
      3            from-pstn                  default
      4            from-pstn                  default
*CLI>
```

第四章测试系统

1. 首先我们测试两个 FXS 口，也就是此图上的第一，第二口准备两部模拟电话，用电话线分别把右图所指的两口与两部电话机接上。

2. 接下来我们开始配置拨号规则，在命令行中输入：
   cd /etc/asterisk
   vi extensions.conf

```
[root@bogon ~]# cd /etc/asterisk
[root@bogon asterisk]# vi extensions.conf
```

跳转到 extensions.conf 的最后一行，添加如图红框所示的代码：

```
; "core show application <command>" will show details of how you
; use that particular application in this file, the dial plan.
; "core show functions" will list all dialplan functions
; "core show function <COMMAND>" will show you more information about
; one function. Remember that function names are UPPER CASE.

[from-internal]
exten => 111,1,Dial(zap/2)
exten => 111,2,Hangup()

exten => 222,1,Dial(zap/1)
exten => 222,2,Hangup()
```

此段拨号规则表示从通道一口的这个电话拨打号码 111 会使通道二的电话振铃
从通道二口的这个电话拨打号码 222 会使通道一的电话振铃
接下来保存退出，输入命令 *asterisk -r* 进入 CLI ，在 CLI 中输入命令 *reload* ，对刚刚的设置进行重载（以后每次修改完 extensions.conf 都需要对 asterisk 进行 Reload 重载）

```
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=========================================================================
Connected to Asterisk 1.4.21.1 currently running on bogon (pid = 3724)
Verbosity is at least 7
bogon*CLI> reload
```

3. 下面我们就可以拿起电话机来测试刚刚所配置的拨号规则是否生效了，如果成功系统会在 CLI 中输出以下字段：

```
== Parsing '/etc/asterisk/sip_notify.conf': Found
  -- Starting simple switch on 'Zap/1-1'
  -- Executing [111@from-internal:1] Dial("Zap/1-1", "zap/2") in new stack
  -- Called 2
  -- Zap/2-1 is ringing
  -- Zap/2-1 is ringing
  -- Zap/2-1 is ringing
```

上图是从通道一口的这个电话拨打号码 111 的输出

```
  == Spawn extension (from-internal, 111, 1) exited non-zero on 'Zap/1-1'
    -- Hungup 'Zap/1-1'
    -- Starting simple switch on 'Zap/1-1'
    -- Executing [111@from-internal:1] Dial("Zap/1-1", "zap/2") in new stack
    -- Called 2
    -- Zap/2-1 is ringing
    -- Zap/2-1 is ringing
    -- Zap/2-1 is ringing
    -- Zap/2-1 is ringing
```

从通道二口的这个电话拨打号码 222 的输出。
有以上输出的话表示你的 FXO 口工作正常。

4. 剩下的就是 FXO 口的测试了，需要一条 PSTN 的电话线，插入第三口。
5. 配置拨号规则，在命令行中输入：

   cd /etc/asterisk

   vi extensions.conf

```
[root@bogon ~]# cd /etc/asterisk
[root@bogon asterisk]# vi extensions.conf
```

跳转到 extensions.conf 的最后一行，添加如图红框所示的代码：

```
[from-internal]
exten => 111,1,Dial(zap/2)
exten => 111,2,Hangup()

exten => 222,1,Dial(zap/1)
exten => 222,2,Hangup()

[from-pstn]
exten => s,1,Dial(zap/1)
exten => s,2,Hangup()
```

这段拨号规则表示从 PSTN 线路上打进来的电话 FXO 口都使通道一的电话振铃
接下来保存退出，输入命令 *asterisk -r* 进入 CLI ，在 CLI 中输入命令 *reload* ，对刚刚的设置进行重载。

6. 下面我们就可以用别的外线电话拨打这条 PSTN 线的号码来测试刚刚所配置的拨号规则是否生效。

成功呼入如下图输出：

```
    -- Starting simple switch on 'Zap/3-1'
    -- Executing [s@from-pstn:1] Dial("Zap/3-1", "zap/1") in new stack
    -- Called 1
    -- Zap/1-1 is ringing
    -- Zap/1-1 is ringing
    -- Zap/1-1 is ringing
    -- Zap/1-1 is ringing
    -- Hungup 'Zap/1-1'
== Spawn extension (from-pstn, s, 1) exited non-zero on 'Zap/3-1'
    -- Hungup 'Zap/3-1'
```

接下来对第四口(FXO 口)进行同理测试，确认其是否工作正常。

第五章　参考

www.openvox.com.cn

www.digium.com

www.asterisk.org

www.voip-info.org

www.asteriskguru.com